

Program - serverSW.c

```
#include <sys/socket.h>
#include <netinet/in.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <unistd.h>

void main(){
    int s,len;
    char smsg[100],rmsg[100];
    struct sockaddr_in server,client;

    int frame=0, in_frame=0;
    char info[100],str1[100],str2[100];

    s = socket(AF_INET,SOCK_DGRAM,0);
    server.sin_family = AF_INET;
    server.sin_port = htons(3000);
    server.sin_addr.s_addr=htonl(INADDR_ANY);

    bind(s,(struct sockaddr*)&server, sizeof(server));

    len = sizeof(client);

    while(1){

        frame++;
        sprintf(str1,"%d",frame);

        recvfrom(s,rmsg,100,0,(struct sockaddr*)&client, &len);
        if (strcmp(smsg,"quit")==0 || strcmp(rmsg,"quit")== 0)
            exit(0);

        printf("Message from client: %s\n",rmsg);
        sscanf(rmsg,"%d",&in_frame);

        if (in_frame == frame){
            printf("Frame %d Recieved\n",in_frame);
            strcpy(smsg,rmsg);
            sendto(s,smsg,100,0,(struct sockaddr*)&client,len);
        }
        else{
            frame--;
            printf("Frame number mismatch\n");
            strcpy(smsg,"Resend Frame ");
            strcat(smsg,str1);
            sendto(s,smsg,100,0,(struct sockaddr*)&client,len);
        }
    }
}
```

```

    }
    close(s);
}

```

Program – clientSW.c

```

#include <sys/socket.h>
#include <netinet/in.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <unistd.h>

```

```

void main(){
    int c,len;
    char rmsg[100],smsg[100];
    struct sockaddr_in server,client;

    int frame=0,in_frame;
    char info[100],str1[100],str2[100];

    c = socket(AF_INET,SOCK_DGRAM,0);

    server.sin_family = AF_INET;
    server.sin_port = htons(3000);
    server.sin_addr.s_addr=htonl(INADDR_ANY);

    bind(c,(struct sockaddr*)&client, sizeof(client));

    len = sizeof(server);

    while(1){

        frame ++;
        printf(str1,"%d",frame);

        printf("Enter frame number : ");
        gets(smsg);

        sendto(c,smsg,100,0,(struct sockaddr*)&server,len);
        if (strcmp(smsg,"quit")==0 || strcmp(rmsg,"quit")== 0)
            exit(0);

        recvfrom(c,rmsg,100,0,(struct sockaddr*)&server, &len);
        if (strcmp(smsg,"quit")==0 || strcmp(rmsg,"quit")== 0)
            exit(0);

        printf("Message from Sever : %s\n",rmsg);
        sscanf(rmsg,"%d",&in_frame);
    }
}

```

```

    if (in_frame == frame){
        printf("Acknowledgment %d Recieved\n",in_frame);
    }
    else{
        frame--;
    }
}
close(c);
}

```

Output

```

admin_GK@administrator52: ~/Desktop
admin_GK@administrator52:~/Desktop$ gcc clientSW.c
clientSW.c: In function 'main':
clientSW.c:34:9: warning: implicit declaration of function 'gets'; did you mean
'fgets'? [-Wimplicit-function-declaration]
   34 |         gets(smsg);
      |         ^~~~~
      |         fgets
/usr/bin/ld: /tmp/ccWaQKXc.o: in function 'main':
clientSW.c:(.text+0xde): warning: the 'gets' function is dangerous and should no
t be used.
admin_GK@administrator52:~/Desktop$ ./a.out
Enter frame number :
Message from Sever : Resend Frame 1
Enter frame number : 1
Message from Sever : 1
Acknowledgment 1 Recieved
Enter frame number : 2
Message from Sever : 2
Acknowledgment 2 Recieved
Enter frame number : 4
Message from Sever : Resend Frame 3
Enter frame number : 3
Message from Sever : 3
Acknowledgment 3 Recieved
Enter frame number : quit
admin_GK@administrator52:~/Desktop$

```

```

admin_GK@administrator52: ~/Desktop
admin_GK@administrator52:~/Desktop$ gcc serverSW.c
admin_GK@administrator52:~/Desktop$ ./a.out
Message from client:
Frame number mismatch
Message from client: 1
Frame 1 Recieved
Message from client: 2
Frame 2 Recieved
Message from client: 4
Frame number mismatch
Message from client: 3
Frame 3 Recieved
admin_GK@administrator52:~/Desktop$

```